MGI

MESTRADO
Gestão de Informação
Master in Information Management

Grade of membership (GoM) por algoritmos genéticos.

Aplicação na segmentação do perfil de engajamento de discotecas no Facebook

Mauricio Vidotti Fernandes

Dissertação apresentada(o) como requisito parcial para obtençãodo grau de Mestre em Gestão de Informação

Dissertation presented as partial requirement for obtaining the Master's degree in Information Management

UNIVERSIDADE NOVA
ISEGI Estatística e Gestão de Informação

Instituto Superior de Estatística e Gestão de Informação
School of Statistics and Information Management
Universidade Nova de Lisboa

Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

GRADE OF MEMBERSHIP (GOM) POR ALGORITMOS GENÉTICOS.

por

Mauricio Vidotti Fernandes

Dissertação apresentada(o) como requisito parcial para a obtenção do grau de Mestre em Gestão de Informação, Especialização em Business Intelligence e gestão do conhecimento.

Orientador/Coorientador: Leonardo Vanneschi

RESUMO

Muitas vezes é necessário trabalhar com variáveis categóricas, porem há um número restrito de análisesque as abordam. Uma boa técnica de segmentação é a grade of membership (GoM), muito utilizada na área médica, em psicologia e em sociologia. Essa metodologia possui uma interpretação interessante baseada em perfis extremos (segmentos) e grau de pertencimento. Porém o modelo possui grande complexidade de estimação dos parâmetros pormáxima verossimilhança. Assim, neste trabalho propõe-se o uso de algoritmos genéticos para diminuir a complexidade e o tempo de cálculo, e aumentar a acurácia. A técnica é nomeada de Genetics Algorithms grade of membership (GA-GoM). Para averiguar a efetividade, o modelo foi primeiramente abordado por simulação – foi executado um experimento fatorial levando em conta o número de segmentos e variáveis trabalhadas. Em seguida, foi abordado um caso prático de segmentação de engajamento em redes sociais. Os resultados são superiores para modelos de maior complexidade. Conclui-se, assim, que é útil a abordagem para grandes bases de dados que contenham dados categóricos.

PALAVRAS-CHAVE

Conjuntos difusos; grade of membership (GoM); modelos difusos; segmentação; meta modelos; algoritmos genéticos; redes sociais; Facebook;Social analytics; Social mining; Social media.

ABSTRACT

Sometimes it is necessary to work with categorical data, but the tools to cluster categorical data are few. One good methodology is grade of membership (GoM) based on fuzzy set used in many areas such as medical science, psychology and sociology. The model has useful concepts like extreme profiles (or classes) and belongs degree (or partial memberships). However, it has ahigh complexity to calculate the parameters by the likelihood. Here it is proposed the use of Genetics Algorithms (GA) for deal with computer performance and accuracy. This method was name Genetics Algorithms grade of membership (GA-GoM). To verify its quality, we compare the needed time and error. Firstwe compared simulatingfractional factorials experiment using number of cluster and variables, after we test for a real clustering of users engagement on night clubs fan pages. The results show a superiority for GA-GoM as time performance and accuracy on the higher complexity models. The model GA-GoM work better with big datasets.

PALAVRAS-CHAVE

Fuzzy sets; Genetics Algorithmic; metamodel; Factorialexperiments; Social analytics; Social mining; Social media

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICEDETABELAS

1. Introdução

As mídias sociais têm estado a gerar cada dia mais informação sobre gostos e interesses de seus utilizadores. Segundo informações veiculadas pelo sitio Expandedramblings, as estimativas do Facebook para maio de 2013 são de que cada utilizador acompanhe em média 40 *fanpages*. Além disso, são estimados cerca de 4,5 mil milhões de *likes* por dia nos conteúdos veiculados na rede.

E as empresas estão a demandar mais integração de sua informação com suas estruturas de bases de clientes. O levantamento, Deloitte University Press (2013), realizado com mais de 2 mil executivos em 2012, revela muito sobre a necessidade e a utilização do Social media. Trata-se de uma ferramenta de marketing cada vez mais importante, como apontada por 80% dos executivos pesquisados. Porém, muitas vezes subutilizada em seu potencial de informação e conhecimento, uma vez que 41% dos executivos consideram que suas empresas coletam pouco ou nada das bases de dados em redes sociais e 43% relatam que analisam pouco ou nada.

O facto de ser relatado por 70% dos executivos revela que há empresas onde se investem recursos para integrar a seu Sistema de CRM dados de redes sociais, a fim de entender seus clientes e atender seus interesses (VanBoskirk et al., 2011).

Com a vantagem de que nas mídias sócias é possível estudar tanto a sua relação com os consumidores como a relação dos seus concorrentes com seus consumidores. Para tal, o trabalho propõe um estudo de caso referente a casas noturnas de Brasília, onde é estudado o padrão de engajamento com os diferentes estabelecimentos.

Tendo dois desafios especiais, as classes não são bem definidas e as variáveis têm que ser trabalhadas categorizadas. Porém, o trato de variáveis qualitativas é um desafio para a estatística, tanto na construção de modelos preditivos como em problemas de classificação de variáveis. Abordagens tradicionais tais como regressões, k-means e Self-Organizing Map (SOM) foram desenvolvidas para variáveis quantitativas, e tratam variáveis qualitativas como variáveis dummys, dicotômicas 0 e 1, o que acarreta alguns problemas de interpretação.

Contudo, um modelo proposto por Woodbury (1970) e Woodburry& Clive (1974) lida com o facto na segmentação ao utilizar pioneiramente a lógica difusa de Zadeh (1965) para segmentação. O modelo em questão, Grade of Membership (GoM), é uma boa metodologia de segmentação para dados categóricos, porém com um algoritmo com interações complexas de cálculo, segundo Yang et al. (2008).

Para Marmelstein (1997), os algoritmos genéticos proporcionam uma alternativa interessante para a estimação por máxima verossimilhança, mas ela mostra limitações para grandes espaços de busca, com a explosão de combinações e, por causa disso, um desempenho computacional fraco. Segundo Park et al. (2005), há muitas publicações de computação evolutiva para construção de cluster.

Existe uma grande quantidade de trabalhos de segmentação por técnicas para criação de cluster por modelos baseados em conjunto difuso. E entre eles, diversos utilizam o modelo GoM.

Apesar disso, foram identificados poucos a pesquisar estudos por esse trabalho, os quais combinam algoritmos difusos de segmentação com algoritmos genéticos. E são citados os autores Park et al. (2005) e Jiang et al. (2013) por Vats. P (2014) em sua revisão bibliográfica, além de Mehdizadeh et al. (2008) e Gao (2003), identificado pelo trabalho. Todos apontam para o Fuzzy c-means, uma técnica de classificação baseada no k-means, destinada a variáveis numéricas.

Este trabalho propõe um modelo alternativo para refinar o processo de cálculo do modelo GoM. Tal modelo visa substituir por algoritmos genéticos a estimativa de parâmetros do modelo feita por máxima verossimilhança, a fim de obter um modelo de maior acurácia e com menor tempo em processamento. Nessa pesquisa é feita uma análise do perfil de engajamento no setor de diversão de Brasília, onde foram selecionadas as fanpages das principais casas noturnas da cidade, cerca a de 25, e foi efetuada a segmentação do perfil de engajamento de todas as pessoas engajadas nas páginas.

1.1. Objetivos

O trabalho visa averiguar a viabilidade de uma nova metodologia de segmentação difusa baseada em GoM, uma expansão por algoritmos genéticos, para facilitar sua aplicação em dados do Facebook. De forma a seguir os seguintes subtópicos:

- Revisar Grade of Membership (GoM)
- Construir e implementar técnica genética de segmentação por Grade of Membership (GoM)
- Implementar computacionalmente o modelo GoM em mesma linguagem para comparação.
- Realizar experimento de comparação da técnica
- Aplicar a um caso de segmentação de mercado alvo em um caso de estudo.

1.2. Relevância

O modelo Grade of Membership (GoM) é uma técnica muito utilizada por diversas áreas, como medicina, psicologia e sociologia (Yang et al., 2008). Seu grande ponto negativo e limitante está no imenso custo computacional e de tempo de processamento por trás da estimação de parâmetros, o que inviabiliza em grande parte problemas de maior complexidade. Com isso, ao propormos uma técnica de melhor estimação com menor tempo de processamento, viabilizamos análises mais complexas. Em especial, por exemplo, para segmentação de dados cadastrais, interesses, perfil de engajamento, entre outros que são corriqueiramente divididos em classes.

O trabalho conta assim com a comprovação experimental do melhor desempenho do modelo. E com um caso prático, onde foi feito um estudo sobre o engajamento dos utilizadores do Facebook. O que torna o exemplo interessante é que dentro dos estudos da mineração de dados nas redes sociais, a maioria foca no mapeamento de Grafos. Há poucos trabalhos que focam na mineração do uso (Fernando, MdGasparMdJohar&Perera 2014). Habbit et al. (2014) aponta que um dos grandes desafios para a ação em comunidades da marca é ter ferramentas analíticas poderosas para estudar o comportamento do consumidor (Habbit et al. 2014).

1.3. Estrutura do trabalho

Este trabalho está organizado com a Seção 2 contendo os conceitos básicos para construção, técnica de conjuntos difusos GoM e algoritmos genéticos, e para um dos modelos foi introduzida a técnica utilizada de experimento fatorial. Na Seção 3 é detalhado o novo método proposto. Na Seção 4 desenvolvem-se as hipóteses e é feita a comparação entre os modelos. Na Seção 5 está a conclusão do trabalho.

2. Referencial teórico

2.1. Conjuntos difusos (Fuzzys sets)

O método estatístico de agrupamento que este trabalho considera mais apropriado para o causo no Social Media é o conjunto difuso. Seu conceito, introduzido por Zadeh (1965), de classes que não são bem definidas, reflete a personalidade e o comportamento das pessoas cuja personalidade é influenciada por diversos perfis, segmentos, e que possuem grau de pertencimento, uma vez que, por vezes, as fronteiras entre uma classe e outra não são bens definidas. “Por isso a ambiguidade não pode ser atribuída a uma dificuldade subjetiva decorrente de um problema de medição ou classificação. Faz parte da natureza do próprio objeto…” Suleman (2009).

O modelo pioneiro de segmentação a usar esse conceito foi o o Grade of Membership (GoM) proposto por Woodburry & Clive (1974) e Woodbury (1970) como referido por Suleman(2009). O objectivo da primeira aplicação era determinar os graus de associativismo de cada paciente em relação a um conjunto K de doenças típicas, a que designaram de perfis puros, conforme Suleman (2009).

Dês de então diversas metodologias baseadas em conjunto difuso foram criadas. Dessas metodologias originaram-se adaptações, ao serem integradas com outras técnicas como estatística Baysena, processos estocásticos etc.

Sendo segundo Yang et al. (2008), Fuzzy c-means (FCM) é o mais estudado, conhecido e usado dos modelos. Também é conhecido como Fuzzy k-means, assim nomeado por ser uma extensão do algoritmo k-means, o qual é baseado em centroide orientado para dados contínuos.

Há também o modelo que possui concepção na lógica difusa *Mixturedistribuition*, mas não é propriamente um modelo misto difuso: entende que o índice pertencimento da classe remete à possibilidade de estar em uma das classes.

2.1.1. Desenvolvimento metodológico do modelo GoM

Para o desenvolvimento metodológico do modelo recorremos a uma resolução matemática exposta por M.-S. Yang et al. (2008), onde temos a base composta por *I* observações de *J* respostas de questões qualitativas com Lj categorias, sendo que cada elemento representado por y_{ijl} possui um valor binário, 0 ou 1. E dimensões:

$$i = 1,2 \dots, I; j = 1,2, \dots, J; l_j = 1,2, \dots, L_j$$

O modelo cria seguimentos λ que foram designados com perfis extremos por Malton et al.(1994), tendo o indivíduo sua posição difusa definida $\vec{g}_i = (g_{i1}, g_{i2}, \dots, g_{ik})$, segundo Suleman (2009) .

O grau de pertencimento é definido pela escore g_{ik} para o perfil k, o qual pode variar entre 0 e 1. Sendo que "0" indica que esses elementos não participam da classe e "1" indica que este elemento é completamente membro da classe (Yang et al., 2008). O posicionamento do indivíduo i tem que está totalmente contemplado em todos os ks, o que implica em:

$$\sum_{k=1}^{K} g_{ik} = 1, \qquad g_{ik} \in \mathbb{R}+, \forall i \tag{2.1.1}$$

A probabilidade λ_{kjl}, do seguimento *k,* está compreendido na variável *j* na categoria *l*, tem que somar 1 para todos subníveis de *j*. Assim temos:

$$\sum_{l=1}^{L_j} \lambda_{kjl} = 1, \qquad \lambda_{ilk} \in \mathbb{R}+, \forall k \tag{2.1.2}$$

Através dos dois parâmetros combinados é possível ter a probabilidade de determinada resposta da matriz pela equação abaixo:

$$P(Y_{ijl} = 1) = \sum_{k=1}^{K} g_{ik}\, \lambda_{kjl} \tag{2.1.3}$$

Função de máximo verossimilhança

Para chegarmos aos parâmetros *g* e λ que melhor estimam a estatística, pode-se recorre à função de verossimilhança, a qual traz como resultado a explicação dos parâmetros. Pode, assim, ser calculado igualando a derivada da função a zero os pontos de máxima.

A função de verossimilhança para os parâmetros *g* e λ do modelo multivariado da distribuição com amostra aleatória simples é:

$$L(g, \lambda) = f_y(y{:}\,\lambda) = \prod_{i=1}^{j} f y_i(y_i{:}\,\lambda) = \prod_{i=1}^{I} \prod_{j=1}^{J} \prod_{l=1}^{L_j} \left(\sum_{k=1}^{k} g_{ik} * \lambda_{kjl} \right)^{y_{jil}} \tag{2.1.4}$$

Para efeito de cálculos pode-se aplicar a log na função, *L(g,λ),* mantendo os pontos de máximo e mínimo da derivada. Temos assim:

$$L_{GoM(g,\lambda)} = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{l=1}^{L_j} ln\left(\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}\right)^{y_{jil}} \quad (2.1.5)$$

Eq. (2.1.5) temos terminado o modelo GoM, como para obtermos os parâmetros da maximização da função L_{GoM} sujeitos a igualdade (2.1.1) e (2.1.2). Consideramos assim oLagrange$\tilde{L_{Gom}}$

$$\tilde{L}_{GoM(g,\lambda)} = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{l=1}^{L_j} ln\left(\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}\right)^{y_{jil}} - w_1\left(\sum_{k=1}^{K} g_{ik} - 1\right) - w_2\left(\sum_{l=1}^{L_j} \lambda_{kjl} - 1\right)$$

Após a primeira derivada teremos:

$$\frac{\partial \tilde{L}_{GoM}}{\partial g_{ik}} = \sum_{j=1}^{J}\sum_{l=1}^{L_j} y_{ijl} \frac{\lambda_{kjl}}{\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}} - w_1 = 0 \quad (2.1.6)$$

$$\frac{\partial \tilde{L}_{GoM}}{\partial \lambda_{ik}} = \sum_{i=1}^{I} y_{ijl} \frac{g_{ik}}{\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}} - w_2 = 0 \quad (2.1.7)$$

$$\frac{\partial \tilde{L}_{GoM}}{\partial w_1} = \left(\sum_{k=1}^{K} g_{ik} - 1\right) = 0 \quad (2.1.8)$$

$$\frac{\partial \tilde{L}_{GoM}}{\partial w_2} = \left(\sum_{l=1}^{L_j} \lambda_{kjl} - 1\right) = 0 \quad (2.1.9)$$

Primeiramente resolvemos w_1 e w_2 baseados na Eqs. (2.1.6) –(2.1.8) e depois substituímos w_1 e $w_{2,,}$ colocamos de volta na Eqs. (2.1.6) e (2.1.7), respectivamente. Após isto temos as condições necessárias para maximizar a função objeto de $GoML_{GoM}$ nas seguintes equações:

$$\sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl} \frac{\lambda_{kjl}}{\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}} - \sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl} = 0 \quad (2.1.10)$$

e

$$\sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl} \frac{g_{kjl}}{\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}} - \sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl} \frac{\lambda_{kjl} g_{kjl}}{\sum_{k=1}^{K} g_{ik} * \lambda_{kjl}} = 0 \quad (2.1.11)$$

Observe que g_{ik} e λ_{kjl} na Eqs. (2.1.10) e (2.1.11) não podem ser resolvidas diretamente. Porém, podemos usar os métodos recursivos, a exemplo do *fixed-point interative*, técnica que a partir de um z retorna uma estimativa melhor para o mesmo, tendo $z^{(t+1)} = f(z^{(t)})$, bastando assim iniciar com $z^{(0)}$ e interagir o número de vezes necessário para uma boa aproximação.

Assim, com base na Eqs. (2.1.10) nos (2.1.11) e a técnica *fixed-point iteration*Yang et al. (2008), apresenta sua simplificação nas Eqs. (2.1.12) e (2.1.13):

$$g_{ik}^{(t+1)} = \frac{\sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl} \frac{{g_{kjl}}^{(t)}\lambda_{kjl}{}^{(t)}}{\sum_{k=1}^{K} g_{ik}*\lambda_{kjl}}}{\sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl}} \qquad (2.1.12)$$

$$\lambda_{kjl}^{(t+1)} = \frac{\sum_{l=1}^{I} y_{ijl} \frac{{g_{kjl}}^{(t+1)}\lambda_{kjl}{}^{(t)}}{\sum_{k\prime=1}^{K} {g_{kjl}}^{(t+1)}*\lambda_{k\prime jl}{}^{(t)}}}{\sum_{j=1}^{J}\sum_{l=1}^{L_l} y_{ijl} \frac{{g_{kjl}}^{(t+1)}\lambda_{kjl}{}^{(t)}}{\sum_{k\prime=1}^{K} {g_{kjl}}^{(t+1)}*\lambda_{k\prime jl}{}^{(t)}}} \qquad (2.1.13)$$

Yang et al. (2008) sintetiza no algoritmo a seguir os passos para resolver a equação recursiva (2.1.12) e (2.1.13):

Passo 1: Defina 1<K<I e defina qualquer e>0

Forneça um valor inicial para $g_{ik}^{(0)}$e $\lambda_{kjl}^{(0)}$ e faça t=0.

Passo 2: Calcule $g_{ik}^{(t+1)}$com $g_{ik}^{(t)}$e$\lambda_{kjl}^{(t)}$ na equação 2.1.12

Passo 3: Atualize $\lambda_{kjl}^{(t+1)}$ com $g_{ik}^{(t+1)}$e$\lambda_{kjl}^{(t)}$ na equação 2.1.13

Passo 4: Calcule para $g_{ik}^{(t+1)}$de $g_{ik}^{(t)}$através da norma$||g_{ik}^{(t+1)} - g_{ik}^{(t)}||$

Se $||g_{ik}^{(t+1)} - g_{ik}^{(t)}||$<e, ENTÃO Pare.

SE NÃO t=t+1 e volte para o passo 2.

Como definido por Yang et al. (2014. P391): “Although the GoM model is a good analysis tool for clustering categorical data, the GoM algorithm has complexiteration calculations.”

2.2. Algoritmos genéticos

Os algoritmos genéticos empregam uma terminologia originada da teoria da evolução natural e da genética.

Um indivíduo da população é representado por um único cromossomo implementado na forma de vetores de atributos, o qual contém a codificação (genótipo, onde cada elemento do vetor é denominado gene) de uma possível solução do problema (fenótipo).

O algoritmo segue basicamente o diagrama abaixo: começa de uma população aleatória inicial. Para repetir o processo que avalia os indivíduos pela função de fitness, atribui a fitness da população pelo indivíduo que melhor se adéqua e aplica os operadores genéticos se a condição de parada não for satisfeita. Repete o processo até ser satisfeita a condição.

Produzem variabilidade através de dois processos básicos: crossover e mutação. Produz-se convergência através da seleção.

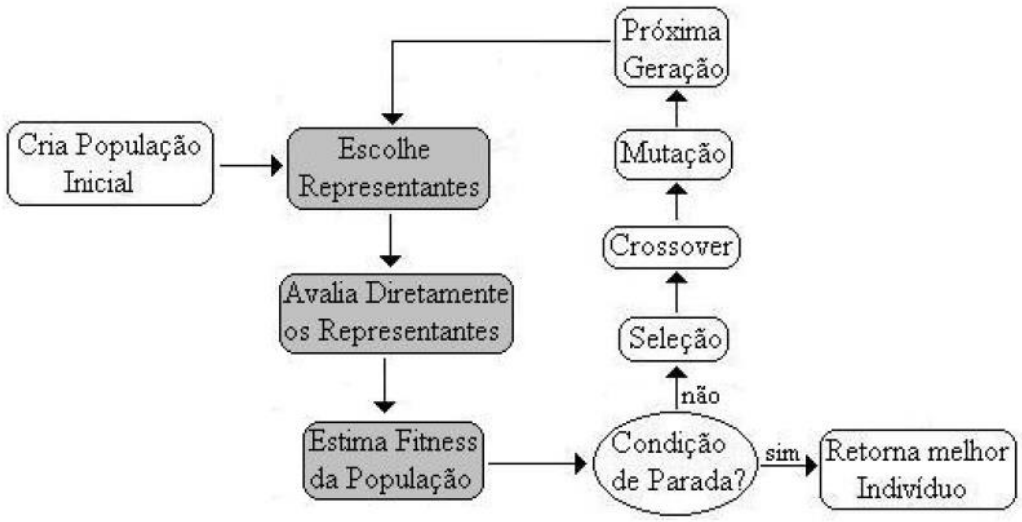

Figura 2-1: Fluxograma de funcionamento de um algoritmo genético

Fonte: Mota Filho (2005)

2.2.1. Função de Fitness

A classificação de indivíduos entre bons e ruins é geralmente feita com base nos respectivos fitnesses (Mota Filho,2005). A função serve para calcular o grau de adaptação do indivíduo na população.

Porém deve haver cuidado na formulação da função objetiva, como define Mota Filho (2005): “Uma formulação errada da função objetivo pode levar à geração da resposta correta para o problema errado.”.

2.2.2. Operadores genéticos

Seleção

Ao contrário, ele seleciona indivíduos relativamente bons em uma população e descarta os restantes, não tão bons.

O que difere os operadores é a natureza da seleção, que pode ser determinística – selecionam-se os melhores, seguindo a ordem da fitness dos indivíduos. Em outro grupo se encontram os operadores de seleção estocásticos, onde a seleção é associada a uma probabilidade de seleção proporcional ao seu fitness.

Independentemente da seleção, é interessante manter uma seleção elitista para garantir que a fitness da população tende sempre a crescer. Uma solução simples, que consiste em sempre manter na próxima geração o melhor indivíduo encontrado na geração atual.

Crossover:

O operador de crossover é o mais característico e o mais utilizado dentro de algoritmos genéticos. A ideia intuitiva por trás do operador de crossover é a troca de informação entre diferentes soluções candidatas (Filho,2005).

A troca de informação por esse operador se dá pela troca de genes através da seleção aleatória de um ponto de corte de dois indivíduos selecionados (pais). Esses geram dois novos indivíduos (filhos) com troca de informação entre ambos. O mais habitual é usar-se um ponto de corte, mas podem ser usados múltiplos pontos até o limite definir se troca gene a gene.

Mutação:

A ideia intuitiva por trás desse operador é a criação da variabilidade extra na população, ou seja, inserir uma pequena perturbação de efeito apenas local (Filho, 2005). Essa perturbação é feita pela alteração aleatória de cada genótipo.

Indivíduos imigrantes:

Uma técnica que evita convergências prematuras é inserir indivíduos imigrantes, novos indivíduos gerados aleatoriamente, na população. Pode ocorrer sempre que a diversidade da população caia abaixo de um limiar, aletoriamente ou mesmo ao longo das gerações.

2.3. Design and analysis of simulation experiments

Segundo Simpson et al. (2001), apesar dos avanços do poder computacional muitas vezes executar repetidamente modelos para comparação de modelos, pode não ser trivial e levar várias horas ou mais.

Para testar os modelos, qualquer que seja o *metamodel,* o analista tem que experimentar, simular com as diversas trocas de parâmetros ou fatores e analisar (Klejinen, 2004). A análise experimental que foi desenvolvida para experimentos físicos pode ser utilizada para o computacional (Kleijnen, 2004; Simpson et al., 2001). Elas são usadas nesse caso para aumentar a eficiência dos testes tradicionais, que variam os parâmetros sistematicamente (Simpson et al., 2001).

2.3.1. Experimento fatorial

Para simulações que precisam levar em conta diversos fatores, uma boa alternativa Klejinen (2004) é a de experimentos fatoriais, uma vez que ela pode levar a interação entre as variáveis em consideração.

O modelo exposto por Mason et al. (2003) visa testar os fatores de influência sobre a média dos experimentos. Sendo μ_{ijkl} o efeito e $_{ijkl}$ o erro aleatório.

$$y_{ijkl} = \mu_{ijkl} + e_{ijkl} \tag{2.3.1}$$

Para cada observação / Montgomery define que o efeito pode ser influenciado pelo modelo, com base nos fatores e suas interações, o que define na equação (2.3.2) para o caso de 3 fatores ou n³:

$$\mu_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + (\alpha\beta)_{ij} + (\alpha\beta)_{ik} + (\beta\gamma)_{jk} + (\alpha\beta\gamma)_{ijk} \tag{2.3.2}$$

O efeito do tratamento e desses fatores é referenciado na equação (2.3.2), sendo o efeito individual do tratamento e de cada fator sobre a média descrito pelas variáveis α_i,β_j e γ_k.

Esses fatores podem ter uma interação conjunta que maximiza ou minimiza seus efeitos quando acontecem em conjunto. Podem ser entre dois, como $(\beta\gamma)_{jk,}$, ou mais fatores $(\alpha\beta\gamma)_{ijk}$. A metodologia visa testar sua significância estatística pelo teste F. Para uma visão mais detalhada, recomenda-se o capítulo 6 do livro de Mason et al. (2003).

Para o melhor desempenho do modelo, o ideal é que os números de experimentos sejam balanceados, iguais para todas as combinações. Há algumas combinações de fatores que por vezes não são exequíveis, seja por tempo, recurso ou algo que impeça. Nesses casos, a saída é fazer o design do experimento incompleto (Mason et al., 2003).

3. Método proposto

A estimação dos parâmetros do modelo GoM pela função de máxima verossimilhança depara-se com um modelo de alta complexidade pelas funções de estimações (2.2.12) e (2.2.13), que são recursivas. Por dependerem de um ponto de início aleatório, podem estar sujeitos a mínimos locais.

Para lidar com tais desafios, o trabalho propõe o uso de algoritmos genéticos e os nomeia por GA-GoM – junção das siglas em inglês GA (GeneticAlgorithm) e da técnica Grade of Membership (GoM).

Tal abordagem consiste em substituir a estimação da matriz de λ expressa pela equação (2.2.12) por um processo de busca baseado em algoritmos genéticos que primeiramente atribui valores aleatórios e depois recorre aos algoritmos de crossover para combinar resultados e convergir para o resultado ótimo.

Para tal, seguem-se as etapas do fluxograma proposto na figura 3-1.

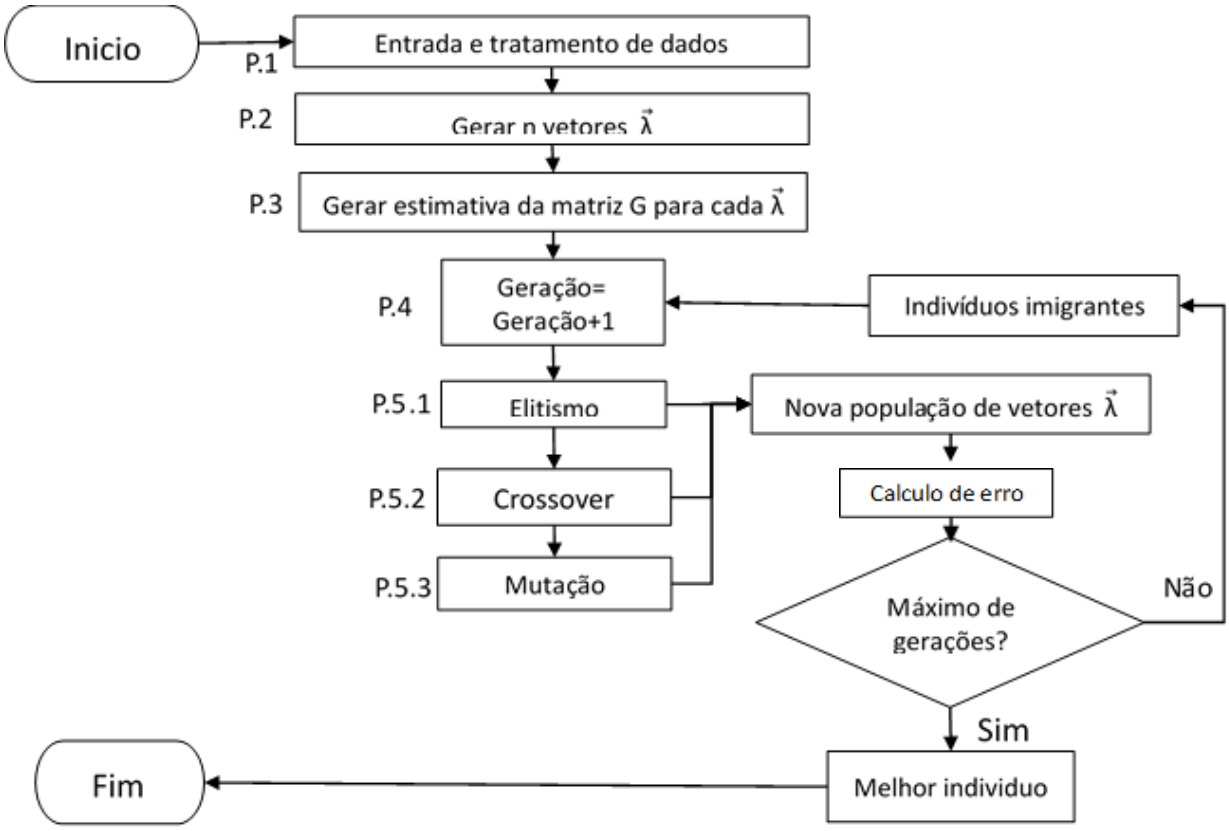

Figura 3-1: Fluxograma de funcionamento de um algoritmo genético adaptado para o modelo GoM.

3.1.1. Preparação das bases de dados

Uma vez que o modelo trabalha com variáveis qualitativas (ordinais ou categóricas), temos que categorizar as variáveis quantitativas de modo a ser viável a aplicação do modelo.

3.1.2. Gerar vetores $\vec{\lambda}$

O modelo é iniciado com a geração aleatória de uma população de Lambda, da qual deve ser estabelecido um número K de cluster para o modelo, o que vai especificar J variáveis e suas respectivas *Lj* subclasses.

Para garantir os pressupostos do modelo, cada variável j da equação 2.1.1, onde se afirmar que o somatório dos pesos do subnível l tem que somar 1, cada gene do modelo passa a ser a variável j, que é composta por um vetor de *l* elementos.

	Var 1, Cluster 1,		Var j, Cluster 1,		Var 1, Cluster k,		Var j, Cluster k,
Cromossomo 1	$\vec{\lambda}_{11}$ 0,5 0,5 0,0	...	$\vec{\lambda}_{1j}$ 0,3 0,4 0,3	...	$\vec{\lambda}_{k1}$ 0,0 0,2 0,8	...	$\vec{\lambda}_{kj}$ 0,1 0,5 0,4
	...						
Cromossomo n	$\vec{\lambda}_{11}$ 0,9 0,05 0,05	...	$\vec{\lambda}_{1j}$ 0,2 0,7 0,1	...	$\vec{\lambda}_{k1}$ 0,8 0,1 0,1	...	$\vec{\lambda}_{kj}$ 0,6 0,3 0,1

Figura 3-2: Representação do genoma do algoritmo genético. O qual trata-se de uma matriz e cada coluna representa um gene.

3.1.3. Estimação da matriz G de pertença

A parte mais complexa consiste em estimar G demonstrando ineficiente o processo de estimação linha-a-linha para trabalhar grandes bases de dados, seja por algoritmos evolutivos ou sistema simplex. Para transpor essa barreira, utilizar a estimação tradicional mostrou-se satisfatório.

3.1.4. Algoritmo genético (Elitismo, Cross Over, Indivíduos imigrantes e Mutação)

Este trabalho utiliza a técnica tradicional para o crossover. A única coisa a diferir é que cada gene também é um vetor, bastando decidir o ponto do crossover aleatoriamente em [1,*J*].

Ponto de crossover

Var 1, Cluster 1,		Var j, Cluster 1,		Var 1, Cluster k,		Var j, Cluster k,
$\vec{\lambda}_{11}$ 0,5 0,5 0,0	...	$\vec{\lambda}_{1j}$ 0,3 0,4 0,3	...	$\vec{\lambda}_{k1}$ 0,0 0,2 0,8	...	$\vec{\lambda}_{kj}$ 0,1 0,5 0,4
$\vec{\lambda}_{11}$ 0,9 0,05 0,05	...	$\vec{\lambda}_{1j}$ 0,2 0,7 0,1	...	$\vec{\lambda}_{k1}$ 0,8 0,1 0,1	...	$\vec{\lambda}_{kj}$ 0,6 0,3 0,1

Figura 3-3: Representação do processo de crossouver com ponto de parada aleatório.

Mutação não foi utilizada no experimento desenvolvido, mas para garantir a diversidade e não conversão prematura foi utilizado Indivíduos imigrantes, onde a cada geração acrescentam-se n vetores de λ novos gerados aleatoriamente.

A fim de garantir a convergência, a metodologia aplica elitismo nos m melhores vetores de λ.

3.1.5. Cálculo da avaliação pela Função de Fitness

A função de *fitness* representa o quanto ele se ajusta e pode ser construída para selecionar o algoritmo que mais acerta ou o que menos erra. Na construção de modelos estatísticos em geral construímos parâmetros que erram menos, investigamos a função de erro para tal. No processo natural de construção do modelo GA-GoM, a função de erro do modelo é expressa por:

$$\frac{\sum_I \sum_J \sum_{Lj} (y_{ijl} - \sum_k \lambda_{kjl} * g_{ijl})^2}{n} = \varepsilon^2 \qquad (3.1.5)$$

4. Desenvolvimento e resultados

Para validar o modelo proposto, o trabalho traz duas abordagens, uma por meio de simulação de casos e outra por uma aplicação prática do caso.

4.1. Simulação

Existem diversas metodologias para a condução do experimento, com o objetivo de averiguar a influência dos tratamentos – no caso, os modelos. Mas para levar em conta e testar também a influênciadas interações, a técnica apropriada é o experimento fatorial.

No caso, cada experimento é uma simulação do modelo com mil observações, sendo os tratamentos as duas metodologias (GoM e GA-GoM) e os fatores referentes ao número de variáveis (5,10) e para o número de cluster (3,4,5,10). Para tal foram feitas duas repetições para cada experimento, totalizando 32 simulações.

A metodologia primeiramente averigua a existência de influência das metodologias (GA-GoM e TR-GoM) e dos fatores (número de variáveis e número de cluster), assim como suas interações para o desempenho da segmentação no tempo e erro. No segundo momento, para o caso onde há influência significativa das interações, são averiguados com maior detalhe pelo teste Tukey, o qual compara a média dois-a-dois para saber onde estão diferenças de médias estatisticamente significativas levando em conta a interação. Pode-se contar com a programação no Anexo 8.3.

Os resultados dos experimentos são descritos na tabela 4.1, onde se pode ter uma visão superficial de que os valores GA-GoM são maiores que os de TR-GoM para 10 variáveis. E menores ou próximos para 5 variáveis, tanto em Erro como no Tempo.

Fatores		Erro		Tempo	
Var	Cluster	GA-GoM	TR-GoM	GA-GoM	TR-GoM
5	3	3,31	3,56	259	179
		3,29	3,52	1332	314
	4	3,31	3,52	1984	178
		3,31	3,26	2074	344
	5	3,29	3,35	2589	172
		3,33	3,43	740	172
	10	3,31	3,38	1257	195
		3,30	3,22	1274	251
10	3	6,83	7,07	2637	5823
		6,81	6,88	3506	2726
	4	6,82	7,11	3981	3982
		6,76	7,08	6883	7152
	5	6,84	6,87	2977	6962
		6,85	7,26	1168	5314
	10	6,82	6,81	4395	5905
		6,83	7,27	1307	5006

Tabela 4-1: Resultado obtido pelas 32 simulações para cada modelo ao variar número de cluster (3,4,5,10) e variar o número de variáveis (5 e 10).

4.1.1. Erro

Na primeira etapa, observa-se na tabela de análise de variância (tabela 4-2), onde são testados com o teste Fa, influência do erro no tipo de modelo, número de variáveis e de (cluster) segmentos e como isso interage com o número de variáveis. Uma vez comprovado como interação do modelo e número de variáveis com um p-valor de 2,10%, tem-se que passar à próxima etapa para entender em maior profundidade essa influência.

Type 3 Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
Modelo	1	25	4.53	0.0434
Var	1	25	1048.98	<.0001
Cluster	1	25	0.08	0.7781
Cluster*Modelo	1	25	0.22	0.6427
Modelo*Var	1	25	6.07	0.0210
Cluster*Var	1	25	0.50	0.4850

Tabela 4-2: Modelo fatorial para fatores de influência no Erro

A figura 4-1 expressa graficamente o teste Tukey, o qual compara cada interação do tipo de modelo (genético ou tradicional) com os níveis de variáveis 5 e 10. Com isso podemos testar as combinações que possuem diferença estaticamente relevante. Assim, temos uma influência positiva do modelo para um conjunto maior de variáveis (10) como o contraponto de não os termos em um pequeno.

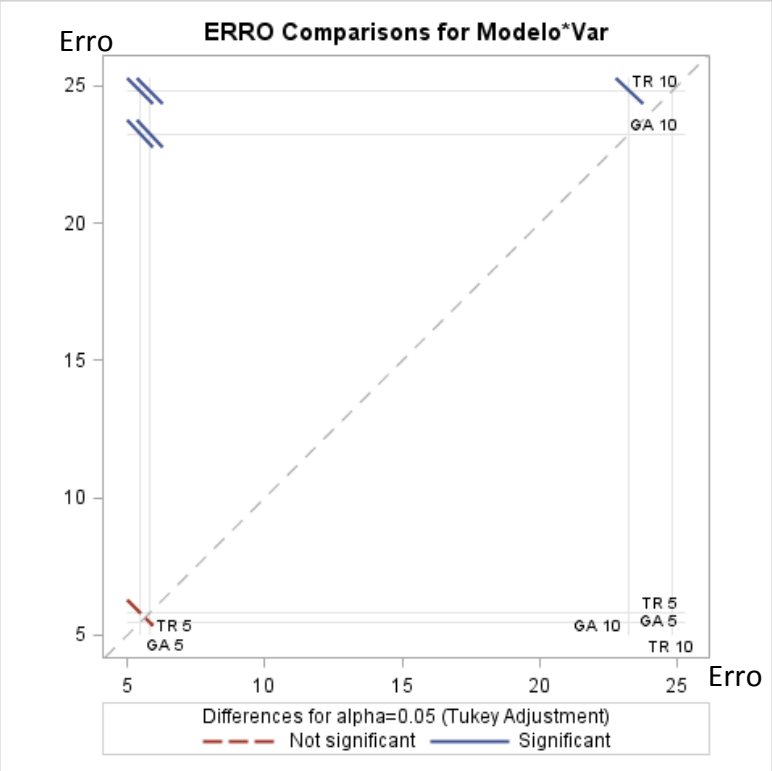

Figura 4-1: Gráfico de comparação de interação de fatores para o Erro

Temos assim expressos na tabela 4-3, numericamente, os resultados gráficos anteriores. Retrata-se a estimativa de erro ajustada pelos efeitos das interações *(L S-Means)* e discriminam-se aquelas estatisticamente diferentes por letras diferentes. No caso, apesar de haver uma melhora nas estimativas com 5 variáveis, não é possível saber se é um efeito aleatório. Já com 10 variáveis a melhora é significativa, como podemos ver pelas letras diferentes.

Tukey Grouping for Modelo*Var Least Squares Means (Alpha=0.05)			
LS-means with the same letter are not significantly different.			
Modelo	Var	Estimate	
TR	10	24.8096	A
GA	10	23.2554	B
TR	5	5.8057	C
			C
GA	5	5.4585	C

Tabela 4-3: Média dos erros das interações entre tratamento (Modelo GA-GoM ou TR-GoM) e fatores de influência (número de variáveis).

4.1.2. Tempo

Na análise do experimento pelo tempo necessário para utilização do modelo, observa-se na tabela 4-4 que o modelo utilizado não tem influência significativa. Porém, repete-se a significância na mesma interação e no tipo de modelo com o número de variáveis. Quando isso ocorre, podem-se desconsiderar os efeitos de menor nível, uma vez que o efeito geral perde importância em contraponto aos efeitos específicos pelas combinações dos subníveis.

Type 3 Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
Modelo	1	25	0.05	0.8292
Var	1	25	12.27	0.0018
Cluster	1	25	0.04	0.8529
Cluster*Modelo	1	25	0.44	0.5127
Modelo*Var	1	25	12.30	0.0017
Cluster*Var	1	25	0.02	0.8940

Tabela 4-4: Modelo fatorial para fatores de influência para o Tempo

Como podemos ver na apresentação gráfica (figura 4-3) do teste de Tukey, as diferenças entre os tratamentos (GA-GoM e TR-GoM) são significantes, para as interações entre os tratamentos aplicados em 10 variáveis, e novamente insignificante para aqueles com 5 variáveis.

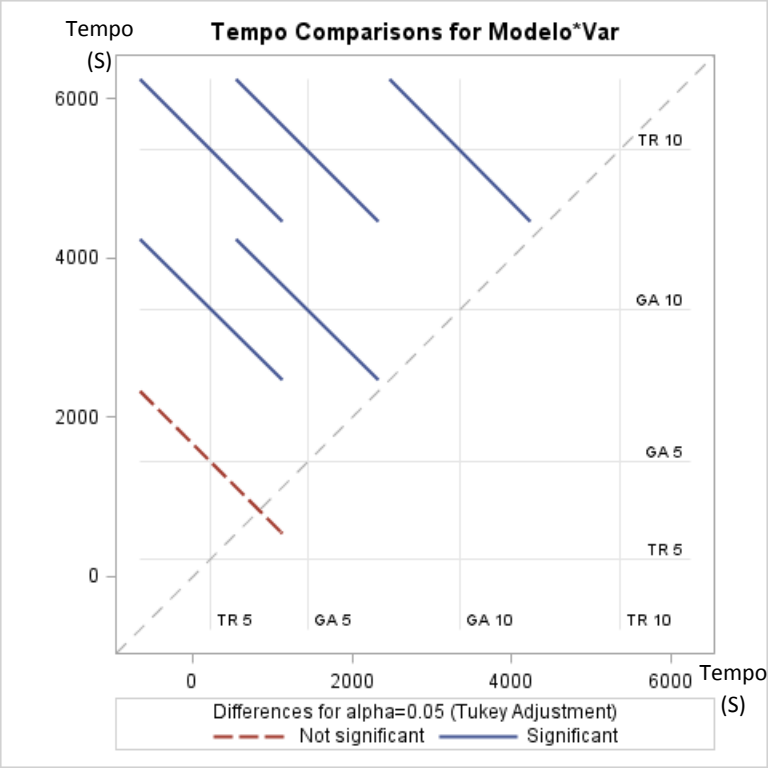

Figura 4-2:Gráfico de comparação de interação de fatores para o Tempo

Quando se olha numericamente, as estimativas das médias pela tabela 4.4 expressam a diferença das médias do modelo GA-GoM maior que o TR-GoM para um número de variáveis igual a 10 e significante. Em contraponto, para 5 variáveis, no tempo a média é numericamente menor, sem relevância estatística. Isso impossibilitaria constar o efeito do tratamento no modelo.

Tukey Grouping for Modelo*Var Least Squares Means (Alpha=0.05)			
LS-means with the same letter are not significantly different.			
Modelo	Var	Estimate	
TR	10	5358.75	A
GA	10	3356.75	B
GA	5	1438.63	C
			C
TR	5	225.63	C

Tabela 4-5 Média dos tempos das interações entre tratamento (Modelo GA-GoM ou TR-GoM) e fatores de influência (número de variáveis).

A diferença, apesar de não significativa, é que o modelo com baixa complexidade tem estimativas bem superiores, como vemos na figura 4-4, porém não é possível afirmar pela grande margem de erro.

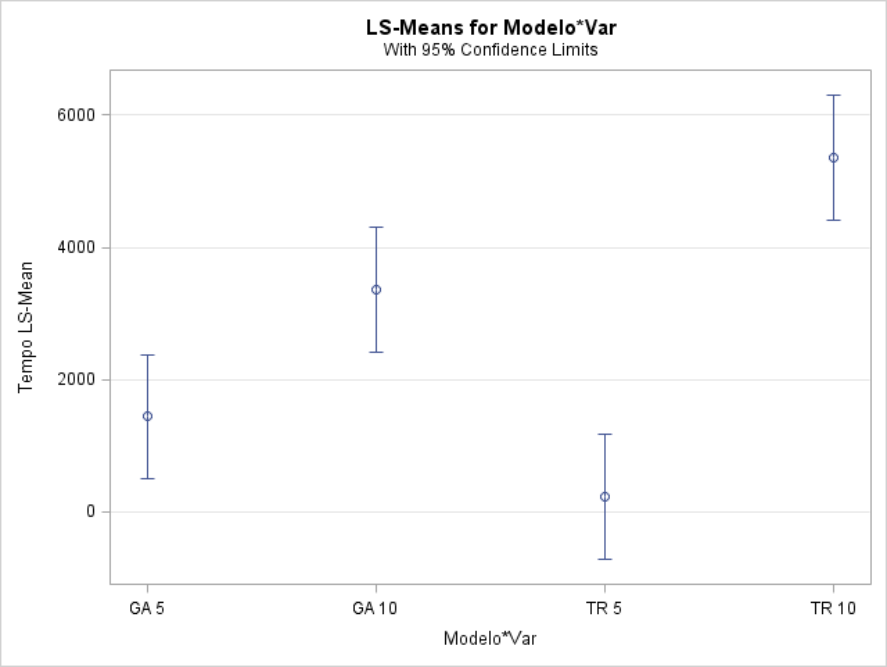

Figura 4-3: intervalo de confiança para médias das interações do tempo

Isso nos leva a concluir que ao aumentarmos o número de variáveis que serão segmentadas os resultados obtidos com o uso de algoritmos genéticos são melhores - tanto em questão de tempo de processamento como em erro de estimação.

4.2. Caso prático

O mercado de discotecas é extremamente concorrido e de risco, pois promover uma festa com pouca adesão pode dar prejuízo. O principal meio de divulgação é pelas redes sociais, em especial o Facebook, uma vez que a escolha do local aonde ir tem uma influência social muito grande e depende muito da recomendação para atrair clientes. Esta aplicação estuda o engajamento dos utilizadores conforme o perfil das casas noturnas que acompanha.

É importante segmentar o perfil de comportamento das pessoas que divulgam discotecas, tanto pela opção "gosto" do Facebook como por comentar na página, uma vez que fazem a informação das *fanpages* repercutir aos seus amigos. Com isso podemos saber quem são as pessoas que estão a interagir, por exemplo, com páginas de discotecas da mesma região ou de outra região. E, assim, saber o perfil de gosto e de engajamento que poderá ser repassado aos *promoters,* representantes comerciais, as pessoas que podem vir a trazer maior retorno para o estabelecimento.

Esse universo, em 2014, totalizou 50.296 pessoas. Por questões de desempenho foram retiradas de uma amostra de 1.550 pessoas dividida em 3 estratos pelo nível de engajamento com a discoteca de referência (Asiático Club), sendo 50 pessoas com um nível de engajamento alto, 500 com nível de engajamento baixo e 1.000 sem engajamento. Para o bom funcionamento do modelo construído por uma amostra, foi importante garantir e manter os sub níveis da casa noturna de referência, uma vez que tais perfis provavelmente não seriam computados em uma amostra padrão por serem muito pequenos. É possível a visualização na tabela 4-5.

ENGAJAMENTO	
ALTO	0,33%
BAIXO	3,76%
NADA	95,91%

Tabela 4-6 Proporção da dispersão das classes de engajamento no Asiático Club

Para a segmentação definimos o perfil das pessoas por seu engajamento com as *fanpages* das 25 principais casas noturnas de Brasília. Foram definidas com isso variáveis referentes ao perfil de engajamento e as características das casas noturnas engajadas.

4.2.1. Os dados

Assim temos as variáveis abaixo para definir os utilizadores:

1) Tipo de engajamento da casa noturna (Asiático) e definição segundo Malthouse et al (2013).
    a. Baixa inciativa de engajamento: quando o utilizador apenas clica like (gosto) no conteúdo recebido.
    b. Maior iniciativa de engajamento: quando o utilizador escreve um comentário na página.
    c. Há também a hipótese de não haver engajamento.

2) Após a frequência de dias em que os utilizadores gostam de algum conteúdo: definida como baixa quando apresenta engajamento apenas um dia, média quando poussi de X a Y, e alta acima de Z
3) Após a frequência de dias em que os utilizadores comentam algum conteúdo: definida como baixa quando apresenta engajamento apenas X dia e alta acima de Z. Existe a possibilidade de não haver comentários.
4) A quantidade de páginas que engajou, conforme as três categorias, apenas uma página, até 3 páginas e mais de 4 páginas.

As casas noturnas foram classificadas conforme estilo, localidade e preço. Com isso observamos:

1) O estilo principal de músicas das casas noturnas que o utilizador engajou, definido em duas categorias gerais: Nacional, Internacional ou Variado (caso o utilizador tenha aderido a casas com músicas nacionais é definido como Variado).
2) Localidade é definida pela região central (Plano Piloto) ou entorno da cidade. Caso a pessoa tenha uma posição de engajamento com ambas será classificada como tal.
3) Preços das casas noturnas são segmentados em duas categorias com valores para a entrada masculina superiores a R$ 40,00 (por volta de ¢10,00) e as que cobram valor inferior a esse. Caso a pessoa tenha uma posição de engajamento com ambas, serão classificadas como tal.

Para mais detalhes do comportamento dos dados, recorrer ao Anexo 8.1.

Análise de correlação das variáveis

Um bom modo de entender a facilidade de redução do modelo é observar a correlação entre as variáveis. Quanto mais correlacionadas, mais próximas e menos variabilidade para ser sintetizada pelo modelo, sendo assim mais difícil a redução de dimensões. A tabela 4-6 expressa a correlação de Cramer's V das variáveis – vemos que as correlações são medianas ou baixas.

	Asiático	Gosto	Comentário	Páginas	Estilo	Local	Preço
Asiático	1,00	0,21	0,44	0,30	0,63	0,36	0,59
Gosto	0,21	1,00	0,06	0,63	0,24	0,48	0,45
Comentário	0,44	0,06	1,00	0,04	0,06	0,07	0,08
Páginas	0,30	0,63	0,04	1,00	0,36	0,59	0,58
Estilo	0,63	0,24	0,06	0,36	1,00	0,53	0,54
Local	0,36	0,48	0,07	0,59	0,53	1,00	0,65
Preço	0,59	0,45	0,08	0,58	0,54	0,65	1,00

Tabela 4-7: Matriz de correlação de Cramer's V das variáveis na amostra

4.2.2. Número de cluster

Conforme a teoria de componentes principais proposta por Person, que decompõe a base de dados em componentes que concentram o máximo da variabilidade da base sequencialmente nos componentes sobre a restrição da independência. Sendo assim, considero um bom referencial alvo para alvo para o erro. Pela técnica, de acordo com software SAS, um bom número de vetores para reduzir a matriz de dados é 4, que explica 70% da variabilidade do modelo.

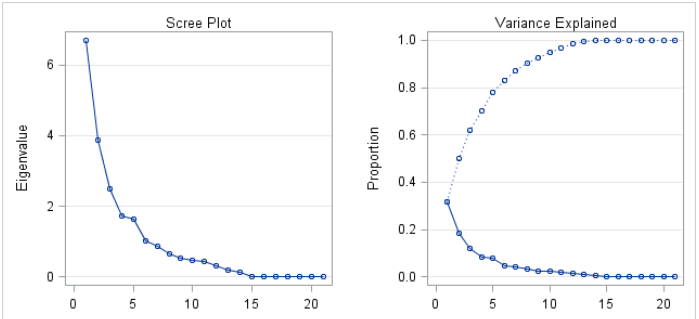

Figura 4-4: Variação explicada pela os fatores do modelo de componentes principais.

4.2.3. Comparação de desempenho dos modelos

Para compararmos o desempenho dos modelos é necessário compararmos os erros de estimação e o tempo despendido para tal.

Erro dos modelos

Os modelos possuem patamar próximo, com uma ligeira melhora de estimação com um erro 2% menor. Ambos são próximos do nível do erro calculado pelo modelo de componentes principais, com erro de 4,8, sendo apenas 15,8% para GA-GoM e 18,3% TR-GoM, que claro não conta com a mesma facilidade de compreensão dos resultados.

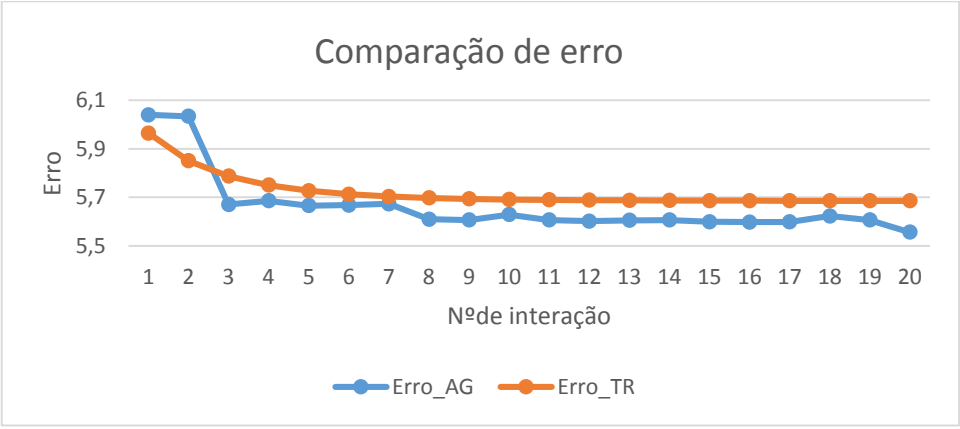

Figura 4-5: Gráfico de comparação de erro dos modelos de GA-GoM e TR-GoM pela interação

Obs: as elevações de Erro_GA ocorrem porque a cada interação é estimada uma nova função de pertence, o que pode ocasionar pequenas elevações do erro do modelo.

Como podemos ver na figura 4-7, o tempo de execução do modelo proposto é muito inferior, mesmo ao se considerar que o modelo GA-GoM converge com 11 interações e

um total de 1 hora e 12 minutos de execução, e o modelo TR com 6 interações e 3 horas e 3 minutos – um pouco menos de um terço do tempo.

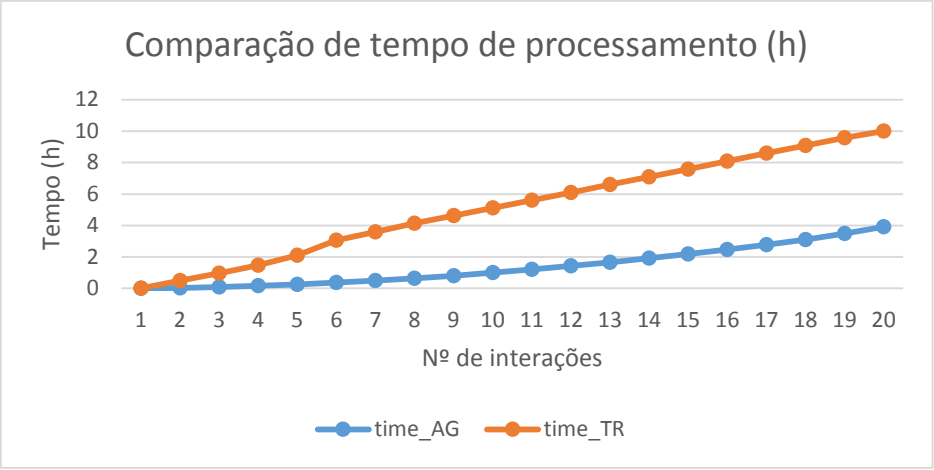

Figura 4-6: Gráfico de comparação de tempo dos modelos de GA-GoM e TR-GoM pela interação

4.3. Estimação de pertencimento nos segmentos (Lambda)

4.3.1. Resultados TR-GOM

A estimação pelo método tradicional (tabela 4.8) é mais sensível a máximos locais. No caso, o conjunto de segmentos encontrados ainda não representa as categorias mais recorrentes que possuem um perfil de engajamento baixo, como é possível ver na tabela de dispersão do Anexo 8.1. Porém, a boa representação das outras variáveis garante um erro baixo.

	Asiático (Alto, Baixo, Nada)	*Freqlike (Alto, Médio Baixo)*	*Freq comente (Alto, Baixo, Nada)*	*Nº de pag (+ de4, 3 ou 2, 1)*	*Musica (Internacional Brasileira, Ambas)*	*Local (Centro, Entorno, Ambos)*	*Preço (<R$40, >R$40, Ambos)*
λ_1	17% 36% 47%	61% 28% 41%	50% 14% 36%	2% **72%** 26%	46% 1% 53%	3% 26% **71%**	**83%** 0% 16%
λ_2	9% **81%** 10%	1% 41% 58%	**88%** 6% 5%	32% 8% 61%	16% 64% 20%	**96%** 3% 1%	**85%** 9% 7%
λ_3	**78%** 21% 1%	**95%** 4% 1%	**40%** **55%** 5%	**28%** **63%** 9%	**53%** 17% 28%	37% **42%** 21%	**61%** 18% 21%
λ_4	55% 1% 44%	**69%** 31% 0%	**96%** 1% 3%	7% 28% 65%	42% 11% 56%	56% 34% 9%	33% **67%** 0%

Tabela 4-8: Composição dos perfis extremos calculado pelo modelo GA

4.3.2. Resultados GA-GOM

Por sua vez, o modelo obtido com algoritmo genético menos suscetível a máximos locais possui maior consistência. A tabela 4-8 representa a composição dos perfis extremos, os segmentos.

	Asiático (Alto, Baixo, Nada)	*Freqlike (Alto, Médio Baixo)*	*Freq comente (Alto, Baixo, Nada)*	*Nº de pag (+ de4, 3 ou 2, 1)*	*Musica (Internacional Brasileira, Ambas)*	*Local (Centro, Entorno, Ambos)*	*Preço (<R$40, >R$40, Ambos)*
λ_1	38% 13% 49%	15% 28% 41%	19% 38% 43%	60% **24**% 16%	31% 47% 22%	**98**% 2% **0**%	**96**% 1% 3%
λ_2	7% **30**% **63**%	25% 14% 61%	**17**% 45% 37%	5% 7% 88%	38% 55% 8%	**66**% 16% 18%	**4**% **87**% 10%
λ_3	**64**% 29% 7%	**19**% **47**% 34%	**51**% **31**% 18%	21% 12% **68**%	**45**% 21% **34**%	10% **65**% 25%	**34**% 11% **55**%
λ_4	34% 20% 46%	**28**% 46% 26%	**46**% 15% 40%	**52**% **48**% 1%	65% 4% 31%	60% 21% 19%	31% **11**% **58**%

Tabela 4-9:Composição dos perfis extremos calculado pelo modelo TR

O modelo generaliza nos perfis extremos da seguinte maneira:

Primeiro segmento, com um perfil de engajamento mediano, mas em várias páginas. Distinguem-se principalmente pelo grande peso a utilizadores que interagem com festas, em geral música nacional no Centro, com ticket baixo.

Segundo perfil, com engajamento que tende a ser baixo, interage com apenas uma festa. Sem perfil de música definido, tendem ser mais no centro e caras. São clientes mais fiéis que podem ser bem rentáveis, merecem especial atenção, porém tendem a interagir menos com o conteúdo.

Terceiro perfil, representando as pessoas que tendem a ter maior engajamento, inclusive de maior interação, ao escrever comentários. Mas em geral tendem a seguir apenas uma página, como a página de referência, Asiático Club. O perfil de estilo está mais em internacional ou variada e muitos em boate do entorno com preços baixos, ou ambos. Traz uma pequena dissonância com o perfil do Asiático no Centro e caro. Mas o fato se dá para fazer a contraposição dos segmentos 1 e 2, que já são majoritariamente no Plano e o 2, majoritariamente caro.

Quarto perfil, de engajamento elevado, como um todo, e presente em várias páginas. Assim, tende a ter um perfil de atuação com o Asiático elevado, preferencialmente englobando eventos com músicas internacionais ou ambas.

O perfil 2 e o 4 podem ser o grande espaço para crescimento do engajamento dos clientes que seguem o mesmo perfil da casa noturna. O 2 pode trazer um retorno financeiro direto maior e o 4, uma repercussão maior.

5. Conclusão

O trabalho propõe um novo algoritmo que combina o modelo de segmentação difuso GoM com algoritmos genéticos GA-GOM, que mostram retorno consistente por experimentos ao simular numericamente os modelos em mesma base de dados. Como em um caso prático de segmentação de perfis de engajamento no Facebook.

Obtém, para bases de maior complexidade, maior acurácia – o que é demonstrado por erros menores e melhor desempenho computacional por tempos menores para problemas com 10 variáveis, assim como no caso prático, que possui um grande poder de síntese de informação: através de 4 perfis tenta explicar 7 variáveis com 3 sub níveis cada, com um total assim de 28 parâmetros a serem estimados.

E é menos sensível a convergências prematuras, como exemplificou a aplicação, que para o modelo GoM tradicional ainda apresentava algumas inconsistências no perfil dos segmentos, mas já havia convergido para um máximo local. Por sua vez, no modelo genético ainda havia espaço para melhorar.

Os resultados vão de encontro aos poucos trabalhos que consideram segmentação fuzzy e algoritmos genéticos, com menos erros e maior resistência a ótimos locais. Mesmo apresentando melhoras robustas no tempo de processamento, é importante a busca por melhorias no desempenho computacional.

Assim, o algoritmo GA-GoM pode ser útil para segmentar dados categóricos, em especial quando comparado a bases de maior complexidade com várias variáveis. O que vai de encontro à crescente demanda de se categorizar o perfil de utilizadores de redes sociais, onde muitas vezes os dados podem ser categóricos.

6. Limitações

O algoritmo proposto leva em conta o modelo GoM. Algumas metodologias mais modernas já foram propostas a partir dele, como por exemplo o fuzzy k-partitions proposto por Yang et al. (2008), entre outros. Contudo, não foram incorporados elementos desses modelos nem comparado seu desempenho, o que pode vir a trazer um melhor desempenho computacional para o modelo.

7. Bibliografia

VanBoskirk, S., Overby, C. S., &Takvorian, S. (2011). U.S. interactive marketing forecast, 2011 to 2016. Cambridge, MA: ForresterResearch. Acessado Dezembro 6, 2014, dehttps://www.forrester.com/US%C3%BEInteractive%20%C3%BE%20Marketing%20%C3%BE%20Forecast%20%C3%BE%202011%20%C3%BE%20To%20%C3%BE%202016/fulltext/-/E-RES59379?docid%20%C2%BC%2059379/

Digital Marketing Ramblings. (2014, Dezembro 01). By the numbers of +200 amazing facebook users statistics [mensagem de blog]. Disponívelem:

Deloitte University Press. (2013). Social business study: Shifting out of first gear. Acessado Dezembro 6, 2014, de http://dupress.com/articles/social-business-study/

Fernando, S. G. S., MdGaparMdJohar&Perera, S.N., (2014). Empirical Analysis of Data Mining Techniques for Social Network Websites. COMPUSOFT, An international journal of advanced computer technology, 3 (2), February-2014 (Volume-III, Issue-II)

W. Glynn Mangold, David J. Faulds (2009). Social media: The new hybrid element of the promotion mix. Business Horizons (2009) 52, 357—365

Malthouse,C. E.,HaenleinM.,Skiera,B.,Wege,E&Zhang M. (2013). Managing Customer Relationships in the Social Media Era: Introducing the Social CRM House. Journal of Interactive Marketing 27 (2013) 270–280

Sabate,F.,Berbegal-Mirabent, J., Cañabate,A. &Lebherz,R.P.(2014). Factors influencing popularity of branded content in Facebookfan pages, European Management Journal 32 (2014) 1001–1011

Choudhury,M. M.&Harrigan,P(2014). CRM to social CRM: theintegration of new technologies into customer relationship management. Journal of StrategicMarketing, 22:2, 149-176, DOI: 10.1080/0965254X.2013.876069http://dx.doi.org/10.1080/0965254X.2013.876069

Brochado, A. & Martins, V. (2001). Bases de Segmentação de Mercado: Classificação e Avaliação. In Actas das *XI Jornadas Luso-Espanholas de Gestão Científica*, Vol. III, Marketing, pp. 97-106.

Vats P.(2014). A Novel Study of Fuzzy Clustering Algorithms for their Applications in Various Domains.The 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (JICTEE-2014)

Yang,S. M., Chiang H. Y., Chen, C. C., Lai Y. C.(2008). A fuzzy k-partitions model for categorical data and its comparison to the GoM model. Fuzzy Sets and Systems 159 (2008) 390 – 405

Malton,K.G.,Woodbury,M.A.&Tolley, H.D. (1994).Statistical Applications Using Fuzzy Sets, Wiley, New York, 1994.

Suleman, A. (2009). Abordagem Estatística de Conjuntos Difusos.(1ª ed) Lisboa: Sílabo.

Pedersen, E. H. E. (2003). Genetic Algorithms for Rule Discovery in Data Mining. Daimi, University of Aarhus, October 2003 http://www.daimi.au.dk/~u971055/

Marmelstein, R. E. (1997). Application of genetic algorithms to data mining, Proceedings of the Eighth Midwest Articial Inteligence and Cognitive Science Conference, The AAAI Press, Dayton, pp. 5357. *http://www.aaai.org/Press/Reports/Conferences/cf-97-01.htmlMota Filho, O. M. F. (2005). Aplicação de Modelos de Estimação de Fitness em Algoritmos Genéticos. Dissertação de mestrado, Universidade de Campinas, Campinas, SP, Brasil.

Park,Han-Saem.,Yoo,Si-Ho&Cho,Sung-Bae(2005). Evolutionary Fuzzy Clustering Algorithm with Knowledge-Based Evaluation and Applications for Gene Expression Profiling. Journal of Computational and Theoretical Nanoscience Vol.2, 1–10, 2005

Bezdek, J. C. & Hathaway, R. J. (1994). Optimization of fuzzy clustering criteria using genetic algorithms, Proceedings of the IEEE Conf. on Evolutionary Computation, Orlando, 2 (1994), 589-594.

Mehdizadeh, E., Sadi-Nezhad, S. &Tavakkoli-Moghaddam R. (2008). Optimization of Fuzzy Clustering Criteria by A Hybrid PSO and Fuzzy C-Means Clustering Algorithm. Iran Journal of Fuzzy Systems, Vol.5 Issue 3, 1-14, 2008

Hongfen J. &Feiyue Y. (2013).An Improved Method of Fuzzy Clustering Algorithm and Its Application in Text Clustering, published in Journal of Information & Computational Science, Vol. - 10-2, pg. no: 519–526, 2013.

Jiang, H., Liu, Y., Ye. F,Xi, H. & Zhu, M. (2013).Study of Clustering Algorithm based on FuzzyC-Means and Immunological Partheno Genetic.Journal of Software, vol. 8, n. 1, 2013.

Jian Gao, Cluster Analysis Based on C-Means and Immune Genetic Algorithm, Computer Engineering, vol. 29, n.12, pp. 65-66, 2003.

Simpson, T. W.,Peplinski, J., Koch, N. P. & Allen, K. J.(2011).METAMODELS FOR COMPUTER-BASED ENGINEERING DESIGN: SURVEY AND RECOMMENDATIONS. Research in Engineerhtg Design.Engineering With Computers, 06/2001; 17(2):129-150.

Kleijnen, P. C. J. (2004).An overview of the design and analysisof simulation experiments for sensitivity analysis. European Journal of Operational Research 16 (2004) 0924-7815

Montgomery, D. C. (2000). Design and Analysis of Experiments, Fifth Edition, New York: John Wiley & Sons, Inc.

Mason, L. M.,Gunst, F. R. & HessL. J. (2003). Statistical Design and Analysis of Experiments with Applications to Engineeringand Science. Second Edition, New York: A JOHN WILEY & SONS PUBLICATION

8. Anexo

8.1. Peso das variáveis aplicação

Distribuição da amostra					
		Asiático			
Variáveis		Alto	Baixo	Nada	Total*
Gosto	Alto	10%	26%	6%	6,5%
	Médio	12%	21%	15%	14,1%
	Baixo	78%	53%	79%	79,5%
Comentário	Alto	10%	0%	1%	0,4%
	Baixo	90%	2%	5%	6,1%
	Nada	0%	98%	94%	93,5%
Nº de paginas	4 ou mais	6%	24%	2%	2,8%
	2 ou 3	28%	32%	15%	17,0%
	Só 1	66%	44%	83%	80,2%
Estilo	Internacional	0%	0%	22%	21,0%
	Brasileira	0%	0%	70%	66,5%
	Variada	100%	100%	8%	12,5%
Local	Centro	78%	55%	49%	47,8%
	Entorno	0%	0%	40%	39,5%
	Ambos	22%	45%	11%	12,7%
Preço	- de R$40,00	0%	0%	86%	81,3%
	+de R$40,00	68%	46%	9%	12,1%
	Ambos	32%	54%	5%	6,6%

Tabela 8-1: Distribuição de variáveis na amostra

8.2. Script SAS Correlação

```
procfreqdata=Asiorder=data;
      tablesasiatico*preco / chisq;
      tables Gosto*preco / chisq;
      tablesComent*preco / chisq;
      tablesPag*preco / chisq;
      tables Estilo*preco / chisq;
      tables local*preco / chisq;
run;

procfreqdata=Asiorder=data;
      tablesasiatico*local / chisq;
      tablesGosto*local / chisq;
      tablesComent*local / chisq;
      tablesPag*local / chisq;
      tablesEstilo*local / chisq;
run;
procfreqdata=Asiorder=data;
      tablesasiatico*Estilo / chisq;
      tables Gosto*Estilo / chisq;
      tablesComent*Estilo / chisq;
      tablesPag*Estilo / chisq;
run;
procfreqdata=Asiorder=data;
      tablesasiatico*Pag / chisq;
      tablesGosto*Pag / chisq;
      tablesComent*Pag / chisq;
run;
procfreqdata=Asiorder=data;
      tablesasiatico*Coment / chisq;
      tablesGosto*Coment / chisq;
      tablesasiatico*Gosto / chisq;
run;

odsgraphicson;

procfactordata=Asiatico1550
priors=smcmsa residual
outstat=fact_all
plots=(screeinitloadingspreloadings loadings);
run;
odsgraphicsoff;
```

8.3. Script SAS Analise de experimento

```
/*ERRO*/

procmixeddata=simumethod=type3;
class modelo var;
model Erro= modelo var cluster cluster*modelo var*modelo var*cluster;
store out1way;
run;
odsgraphicson;
odshtmlstyle=statistical sge=on;
procplm restore=out1way;
lsmeansvar*modelo / adjust=tukey plot=meanplot cl lines;
odsexclude diffs diffsplot;
run; title; run;

                                                    /*Tempo*/

procmixeddata=simumethod=type3;
class modelo var;
model TEMPO= modelo var cluster cluster*modelo var*modelo var*cluster;
store out2way;
run;
odsgraphicson;
odshtmlstyle=statistical sge=on;
procplm restore=out2way;
lsmeansvar*modelo / adjust=tukey plot=meanplot cl lines;
odsexclude diffs diffsplot;
run; title; run;
```

8.4. Algoritmo modelo GA-GoM

```
Gn=list()

erron=list()

K=4

Lambn=list()

Time=list()

Time2=list()

ini = time.time()

J=len(Lj)

#tranformarmatriz teste em Y

Y=list()

fori in range(0,I):

   l1=int()

   l2=int()

Yl=list()
```

```
Yj=list()
delYj[:]
for j in Lj:
l1=l2
l2=l1+j
Yl=teste[i][l1:l2]
Yj.append(Yl)
Y.append(Yj)
#Criar matriz G #G[i][k]
def Cg(I):
global G
   G=list()
fori in range(0,I):
gk=list()
cgk=int()
ccgk=int()
for k in range(0,K-1):
cgk=random.uniform(0,1-ccgk)
ccgk=cgk+ccgk
gk.append(cgk)
gk.append(1-ccgk)
G.append(gk)
#Criar matriz Lambda #Lamb[k][j][l]
def Cl(K):
global Lamb
   Lamb=list()
for k in range(0,K):
lambj=list()
for l in Lj:
```

```
lambl=list()
clk=int()
cclk=int()
for j in range(0,l-1):
clk=random.uniform(0,1-cclk)
cclk=clk+cclk
lambl.append(clk)
lambl.append(1-cclk)
lambj.append(lambl)
Lamb.append(lambj)
#Função de para Calcular G
defgt(n):
i=0
   k=0
globalGn
fori in range(0,I):
for k in range(0,K):
        som1=int()
        som2=int()
        somp1_1=int()
for j in range(0,J):
forlj in Lj:
for l in range(0,lj):
                somp1_1=int()
forkk in range(0,K):
somp1_1=Gn[n][i][kk]*Lambn[n][kk][j][l]+somp1_1
som1=(Y[i][j][l]*G[i][k]*Lambn[n][k][j][l]/somp1_1)+som1
som2=Y[i][j][l]+som2
Gn[n][i][k]=som1/som2
```

```
#Calcular o erro do modelo Y[i][j][l]- (Soma em K de) {G[i][k1]*Lamb[k1][j][l]
def cerro(n):
erro=int()
fori in range(0,I):
for j in range(0,J):
forlj in Lj:
for l in range(0,lj):
estimativa=int()
for k in range(0,K):
estimativa=Gn[n][i][k]*Lambn[n][k][j][l]+estimativa
erro=(Y[i][j][l]-estimativa)**2+erro
global erro
defMenorerro():
global ERRO
globalerron
globalLambn
globalLmax
globalLAMBDAMAX
LAMBDAMAX=list()
   n=int()
forck in range(1,len(Lambn)):
ekr=int()
forcj in range(1,len(Lambn)):
iferron[ck]>erron[cj]:
ekr=1+ekr
ifekr<kmax and n<nkmax:
LAMBDAMAX.append(Lambn[ck])
n+=1
ifekr==0:
```

```
ERRO.append(erron[ck])
Lmax=Lambn[ck]

defCrosover(x,z):
globalLambn
cross=list()
   cross2=list()
   r=random.randint(0,len(Lj))
cross = x[:r]+z[r:]
cross2 = x[r:]+z[:r]
Lambn.append(cross)
Lambn.append(cross2)

# ------------------------------------------------------------------------------
#                    PROCESSO COMPLETO
kmax=3
nkmax=3
NLREP=2
REPG=2
Cg(I)

for n in range(0,NREP):
Gn=list()
erron=list()
LmaxLn=list()
Nl=15
LmaxLmax=list()
LAMBDAMAX=list()
Lmax=list()
```

```
   #CriarLambdas
fornl in range(0,Nl):
Cl(K)
Lambn.append(Lamb)
Nl=len(Lambn)

   #Crossover(Calcular entre:
#  Lambn VRs LAMBDAMAX
iflen(LAMBDAMAX)>0:
fornlrep in range(0,NLREP):
          n=random.randint(0,len(LAMBDAMAX))
          m=random.randint(0,len(Lambn))
if [n,m] not in LmaxLn:
Crosover(LAMBDAMAX[n],Lambn[m])
LmaxLn.append([n,m])

      #os melhores (LAMBDAMAX)
fornlrep in range(0,NLREP):
          n=random.randint(0,len(LAMBDAMAX))
          m=random.randint(0,len(LAMBDAMAX))
if [n,m] not in LmaxLmax:
Crosover(LAMBDAMAX[n],LAMBDAMAX[m])
LmaxLmax.append([n,m])
      #Elitismo
Lambn.append(Lmax)


   #Repetir n vezes
fornl in range(0,Nl):
```

```
#Calcular G para cada Lambda
        #adaptar para NL Lamibidas
for rep in range(0,REPG):
Cg(l)
Gn.append(G)
gt(nl)
      #Calcular erro para cada Lambda
        #adaptar para NL Lamibidas e Gs
cerro(nl)
erron.append(erro)
   #Escolher o melhor Lambda
Menorerro()
Time.append(time.time())
print('GGOM:',n)

fim = time.time()
tempo=fim-ini



ini2 = time.time()
```

8.5. Algoritmo computacional tradicional GoM

```
#Objetivo calcular #matrizez:
   #y[i][j][l]
#g[i][k]
   #Lamb[k][j][l]
#Criar vetor de teste dos modelos
```

```
I=len(teste)
J=len(Lj)
#Criar matriz G #G[i][k]
G=list()
fori in range(0,I):
gk=list()
cgk=int()
ccgk=int()
for k in range(0,K-1):
cgk=random.uniform(0,1-ccgk)
ccgk=cgk+ccgk
gk.append(cgk)
gk.append(1-ccgk)
G.append(gk)
#Criar matriz Lambda #Lamb[k][j][l]
Lamb=list()
for k in range(0,K):
lambj=list()
for l in Lj:
lambl=list()
clk=int()
cclk=int()
for j in range(0,l-1):
clk=random.uniform(0,1-cclk)
cclk=clk+cclk
lambl.append(clk)
lambl.append(1-cclk)
lambj.append(lambl)
Lamb.append(lambj)
```

```
#Função de para Calcular G
i=0
k=0
defgt():
fori in range(0,I):
for k in range(0,K):
        som1=int()
        som2=int()
        somp1_1=int()
for j in range(0,J):
forlj in Lj:
for l in range(0,lj):
                somp1_1=int()
forkk in range(0,K):
                   somp1_1=G[i][kk]*Lamb[kk][j][l]+somp1_1
                som1=(Y[i][j][l]*G[i][k]*Lamb[k][j][l]/somp1_1)+som1
som2=Y[i][j][l]+som2
        G[i][k]=som1/som2
#Função para calcular LAMBDA
def Lt():
   k=int()
   j=0
   l=0
for k in range(0,K):
for j in range(0,J):
forlj in Lj:
for l in range(0,lj):
fori in range(0,I):
                L1=int()
```

```
                L2=int()
                somp2_1=int()
forkk in range(0,K):
                   somp2_1=G[i][kk]*Lamb[kk][j][l]+somp2_1
                L1=Y[i][j][l]*G[i][k]*Lamb[k][j][l]/somp2_1


fori in range(0,I):
for k in range(0,K):
                   som1=int()
                   som2=int()
                   somp1_1=int()
for j in range(0,J):
forlj in Lj:
for l in range(0,lj):
forkk in range(0,K):
                              somp1_1=G[i][kk]*Lamb[kk][j][l]+somp1_1
                           som1=(Y[i][j][l]*G[i][k]*Lamb[k][j][l]/somp1_1)+som1
som2=Y[i][j][l]+som2
                      L2=som1/som2
             Lamb[k][j][l]=L1/L2
#Calcular o erro do modelo Y[i][j][l]- (Soma em K de) {G[i][k1]*Lamb[k1][j][l]
deferrot():
global erro2
erro2=int()
ERRO2=list()
fori in range(0,I):
for j in range(0,J):
forlj in Lj:
for l in range(0,lj):
```

```
estimativa=int()
for k in range(0,K):
estimativa=G[i][k]*Lamb[k][j][l]+estimativa
erro2=(Y[i][j][l]-estimativa)**2+erro2

ERRO2=list()
for n in range(0,NREP):
gt()
Lt()
errot()
ERRO2.append(erro2)
Time2.append(time.time())
print('TGOM:',n)
fim2=time.time()
tempo2=fim2-ini2
```